**Check if a given array is a subset of another array. Duplicate elements can be present.**

Given two arrays, *arr1* and *arr2*, check if *arr2* is a subset of *arr1*.

**Example-1:**

**Input:**

arr1 = {2,4,7,1,5,5}

arr2 = {5,4,2}

**Output:** arr2 is a subset of arr1

**Explanation**: All elements of arr2 are present in arr1.

**Example-2:**

**Input:**

arr1 = {9,3,1,5,2,1}

arr2 = {9,1,1,1}

**Output:** arr2 is not a subset of arr1

**Explanation**: Element 1 is present twice in arr1 and thrice in arr2.

int n = nums2.length();

//hash Map to store frequencies

Map<Integer , Integer > map = new HashMap<>();

// Make count of nums1 bcause of duplicates

for(int i = 0 ; i < nums1.length ; i++){

map.put(nums1[i] , map.getOrDefault(nums1[i] , 0) + 1);

}

for(int i = 0 ; i < n; i++){

// Element is not present

if(!map.contains(nums2[i]) )

return false;

// If element appears more than the original array

if (map.get(nums2[i]) == 0)

return false;

//Reduce the count for each element present in the original array

map.put(nums2[i] , map.getOrDefault(nums2[i] , 0) - 1);

}

return true;

}